

Healthcare Predictive Modeling with Graph Networks

Wade Schulz, MD, PhD

Assistant Professor, Yale School of Medicine

Founder, Refactor Health

LinkedIn: <https://www.linkedin.com/in/wadeschulz/>

Twitter: @wade_schulz

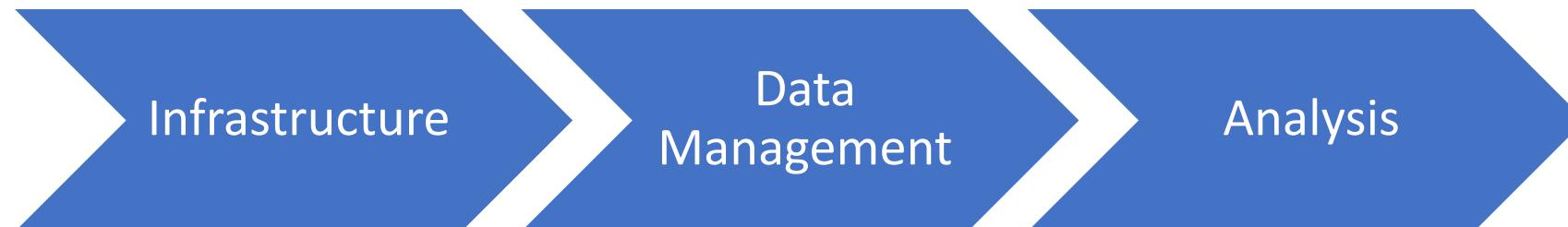


Overview

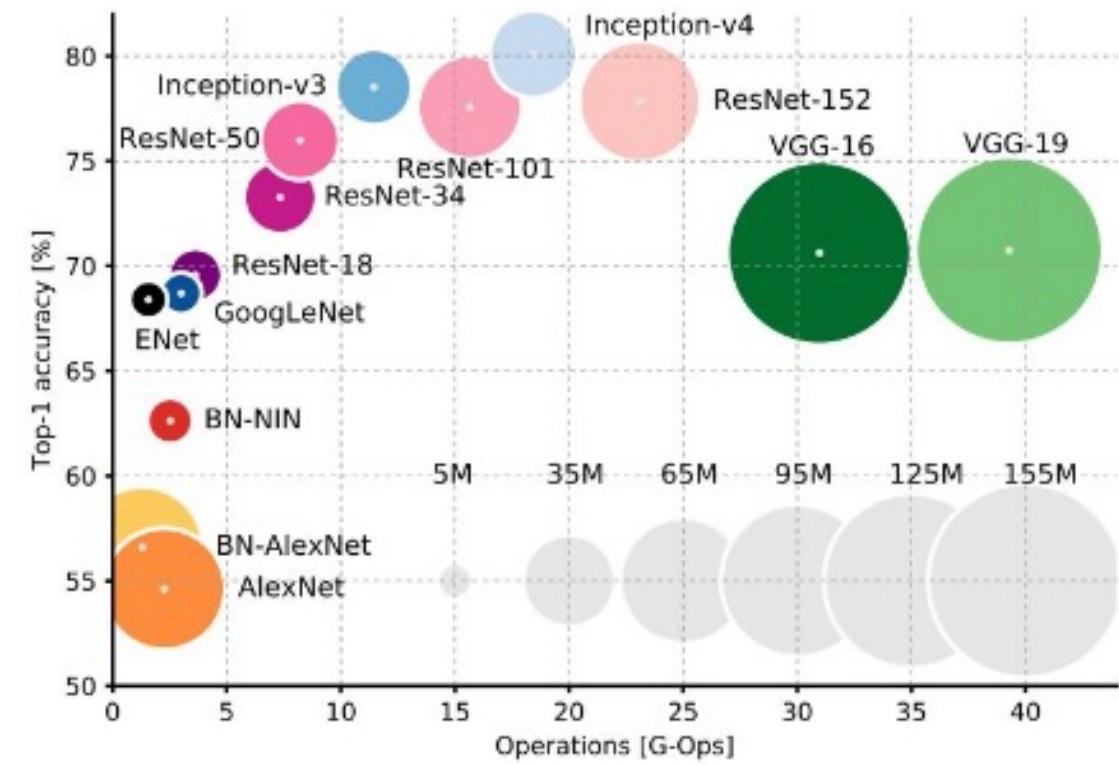
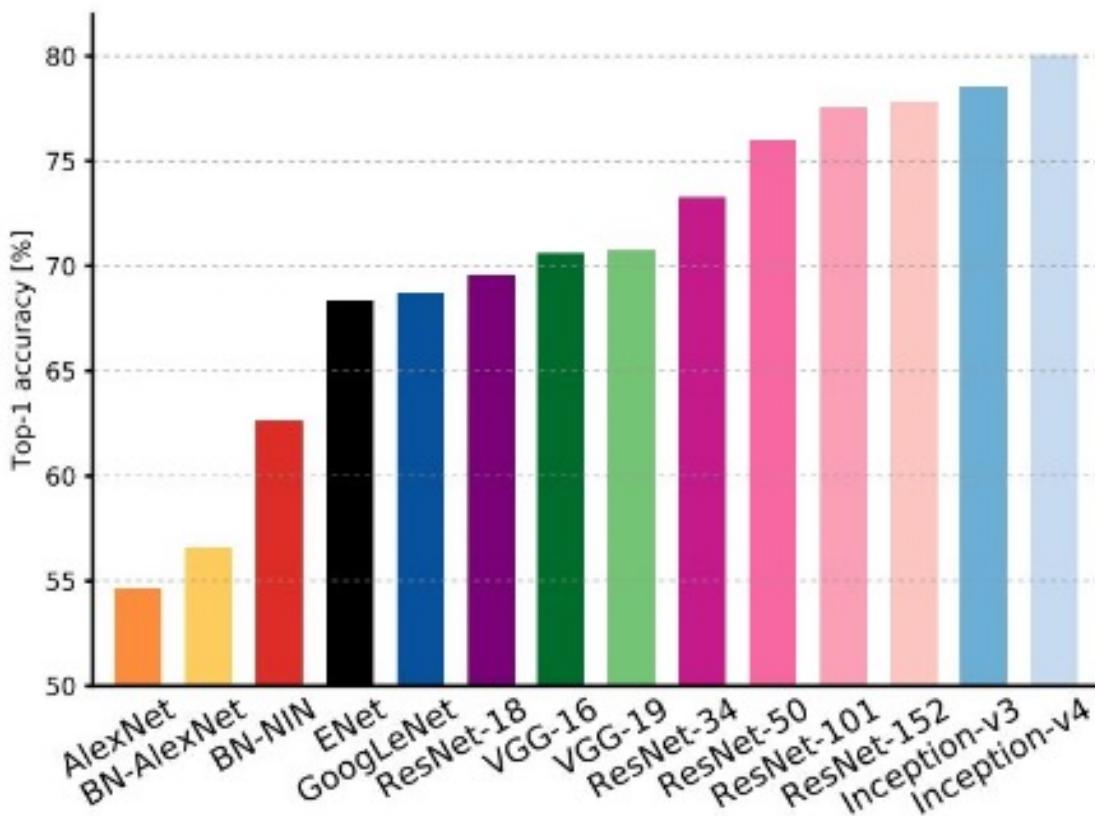
- Why graph networks?
- Limitations of traditional tools when applied to graphs
- Graph models in healthcare and biotech
- Impact of data models on predictive models
- Tools to enable AI with graph models

Computational Healthcare

The interdisciplinary application of innovative data science tools to address health-related questions and problems



Advancements in AI/ML



An Analysis of Deep Neural Network Models for Practical Applications, 2017.

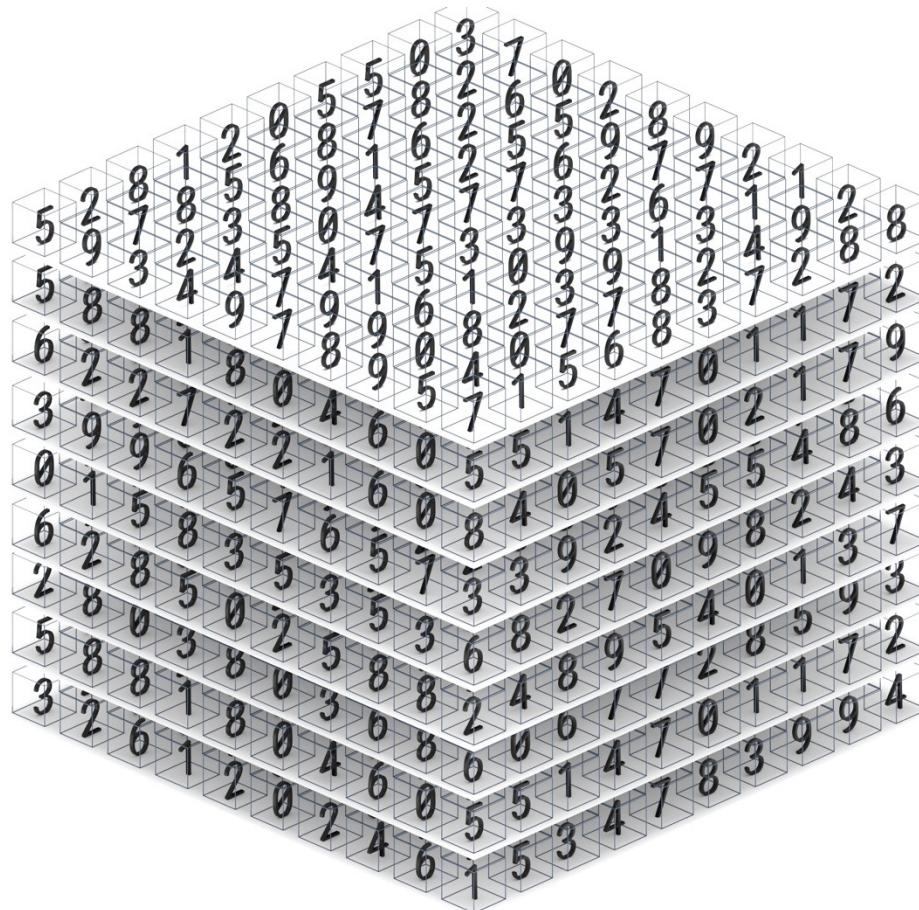
Features + Relationships

0	0	152	244	255	255
0	152	244	255	255	0
0	0	0	152	244	0

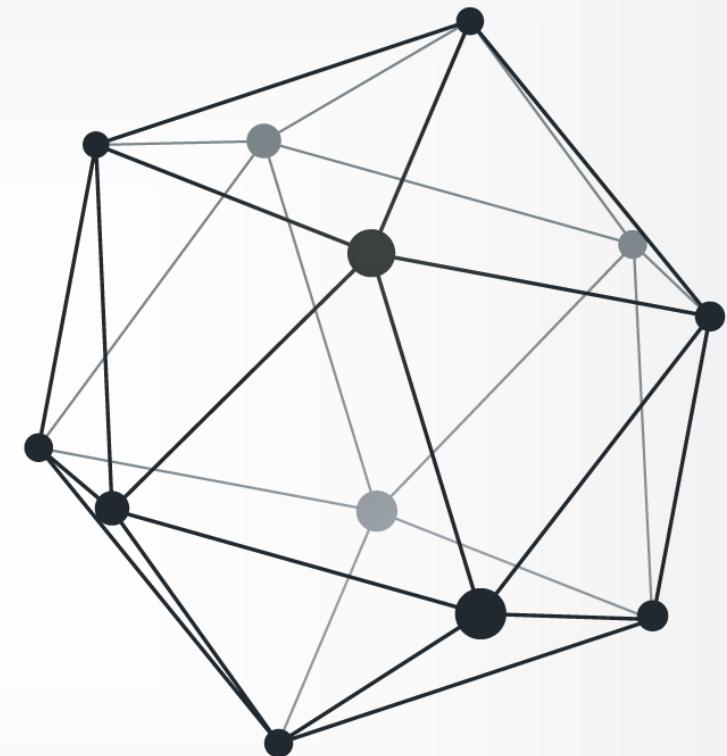
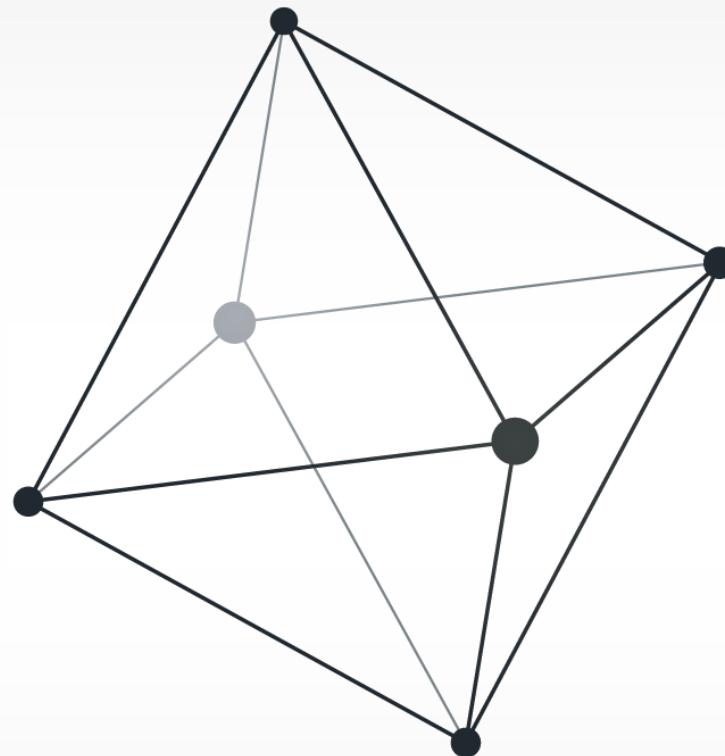
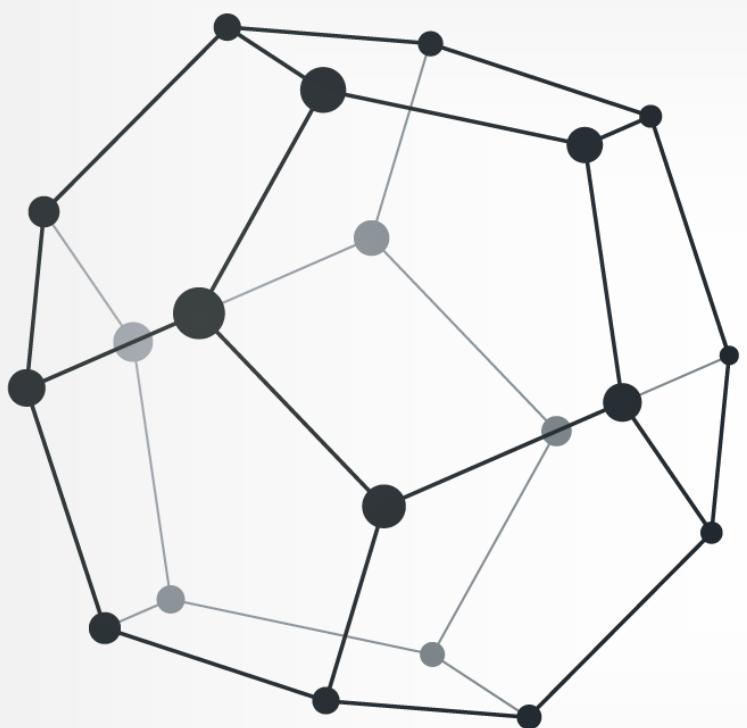
Features + Relationships

0	0	152	244	255	255
0	152	244	255	255	0
0	0	0	152	244	0

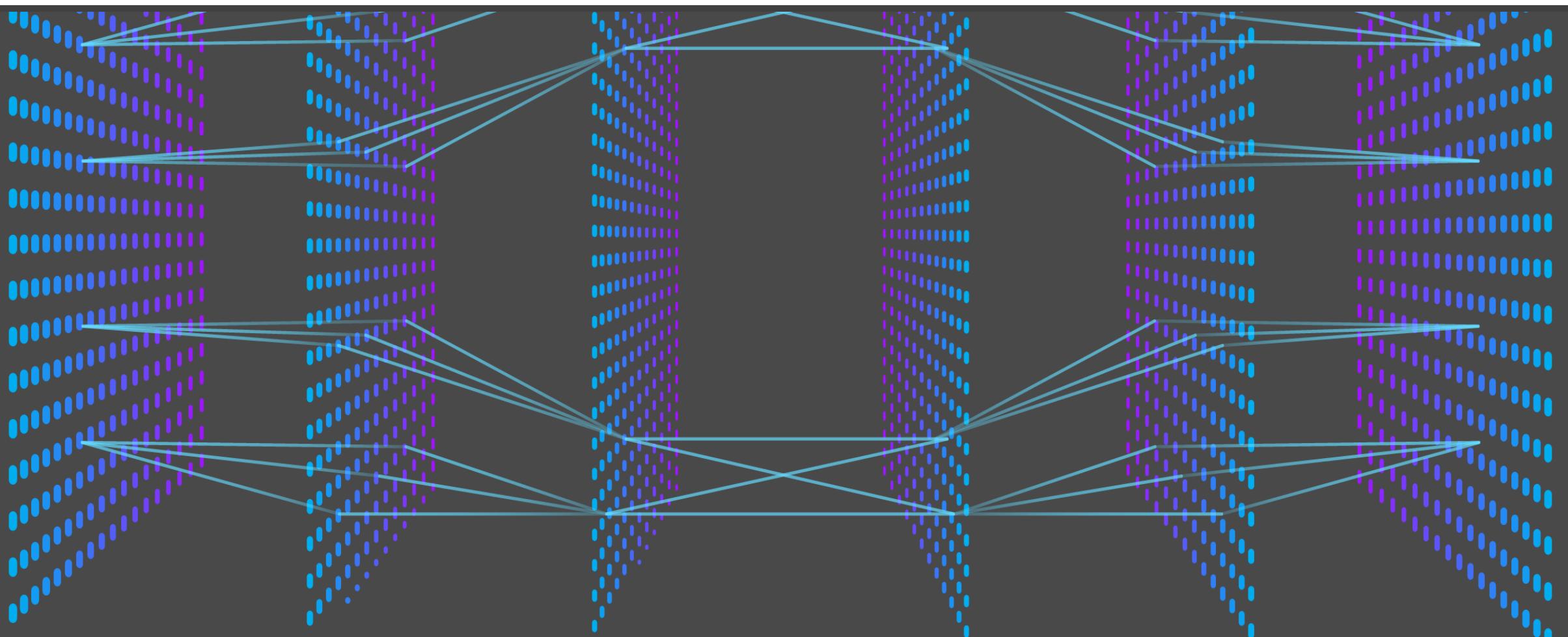
Limited Dimensions



Graphs: *The Next Frontier*

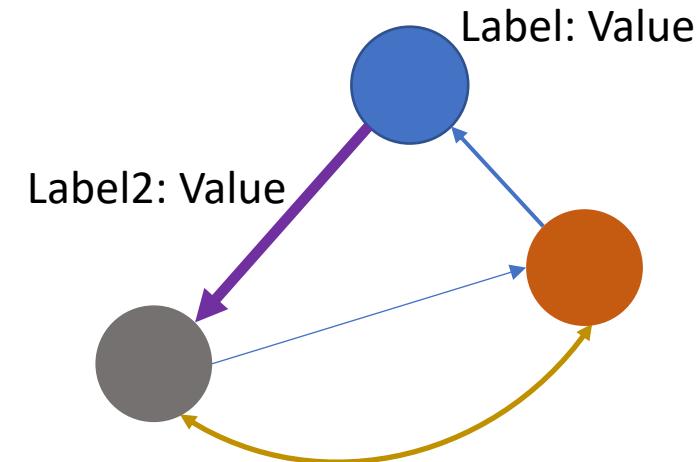
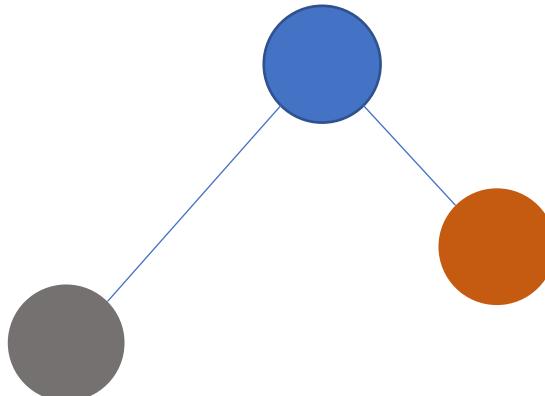


But...how do we learn from a graph?



Learning Frameworks and Graph Networks

- Manual approaches to feature engineering are complex and error prone
- Traditional ML frameworks do not scale efficiently to the size of the graph network when relationships are added



Stanford Graph Learning Workshop

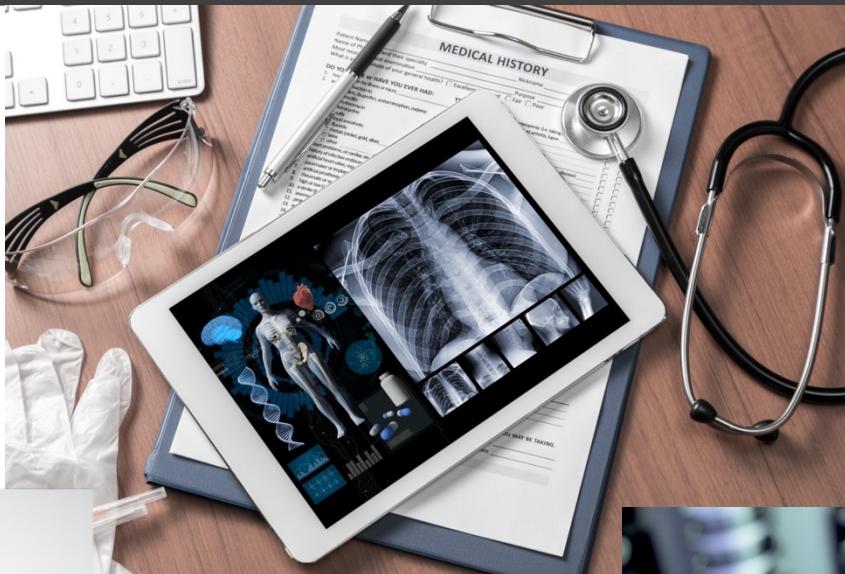


Stanford ENGINEERING | Stanford Computer Forum
Stanford | Data Science

<https://snap.stanford.edu/graphlearning-workshop/>



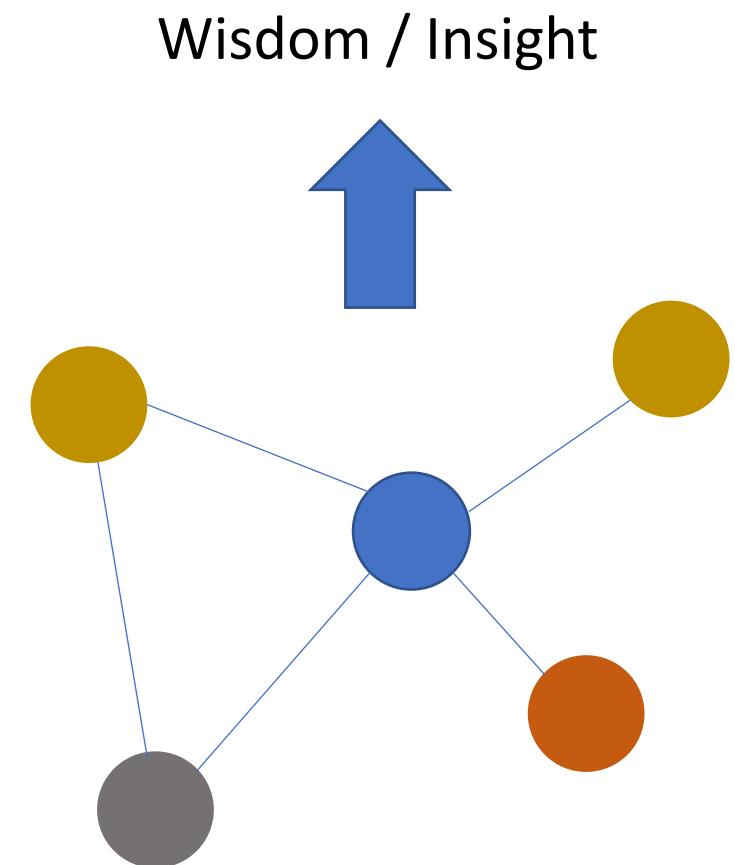
Applications of Graphs in Healthcare



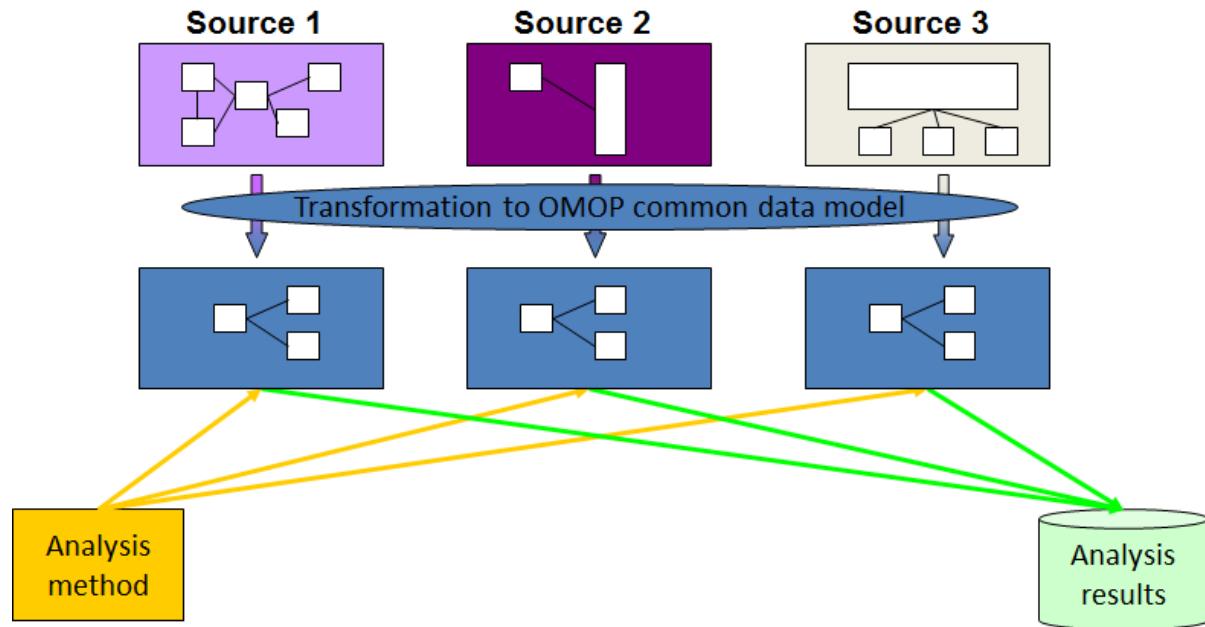
Data to Knowledge to Insights



https://en.wikipedia.org/wiki/DIKW_pyramid#/media/File:DIKW_Pyramid.svg

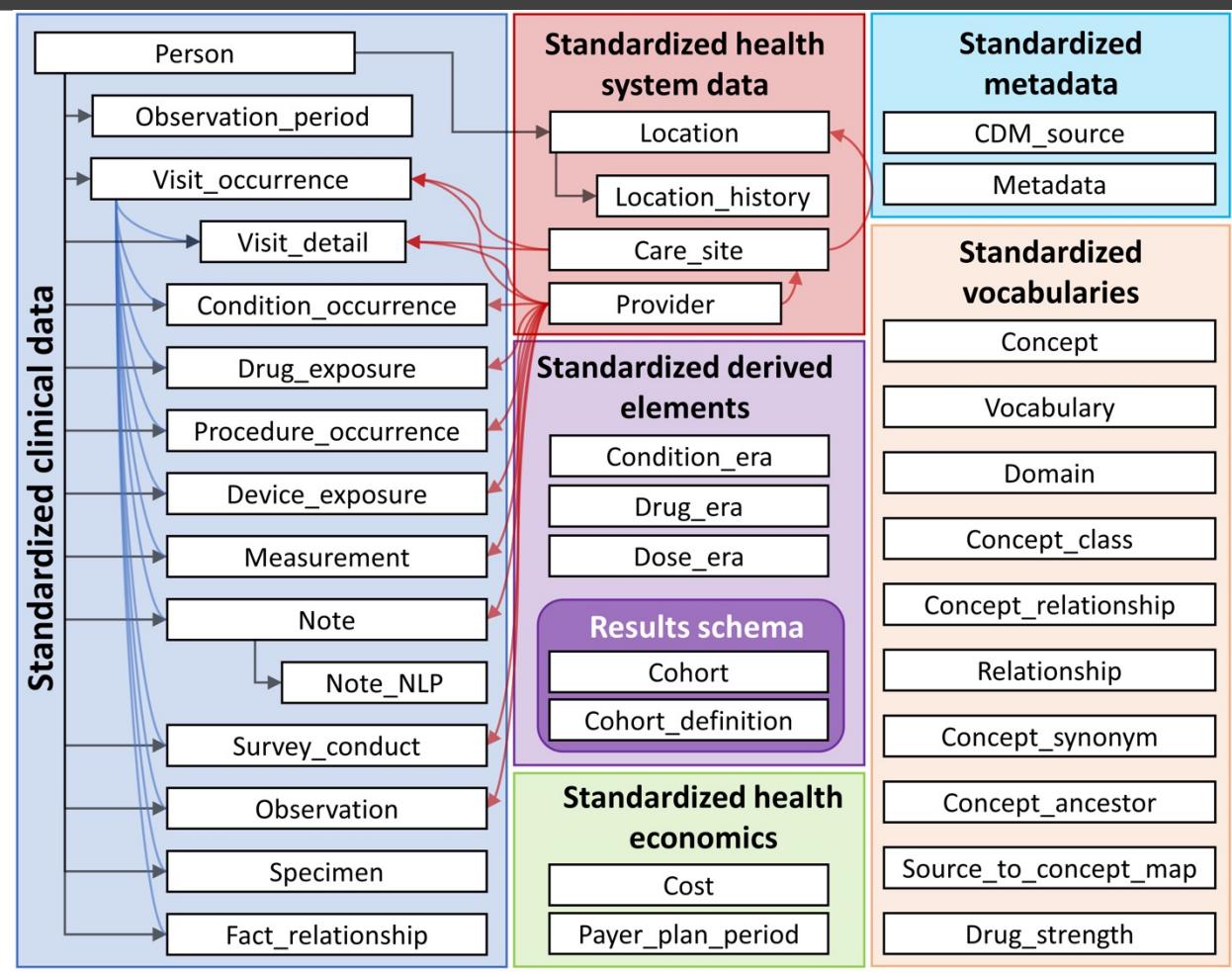


Typical Healthcare Data Models



OHDSI and the OMOP Model

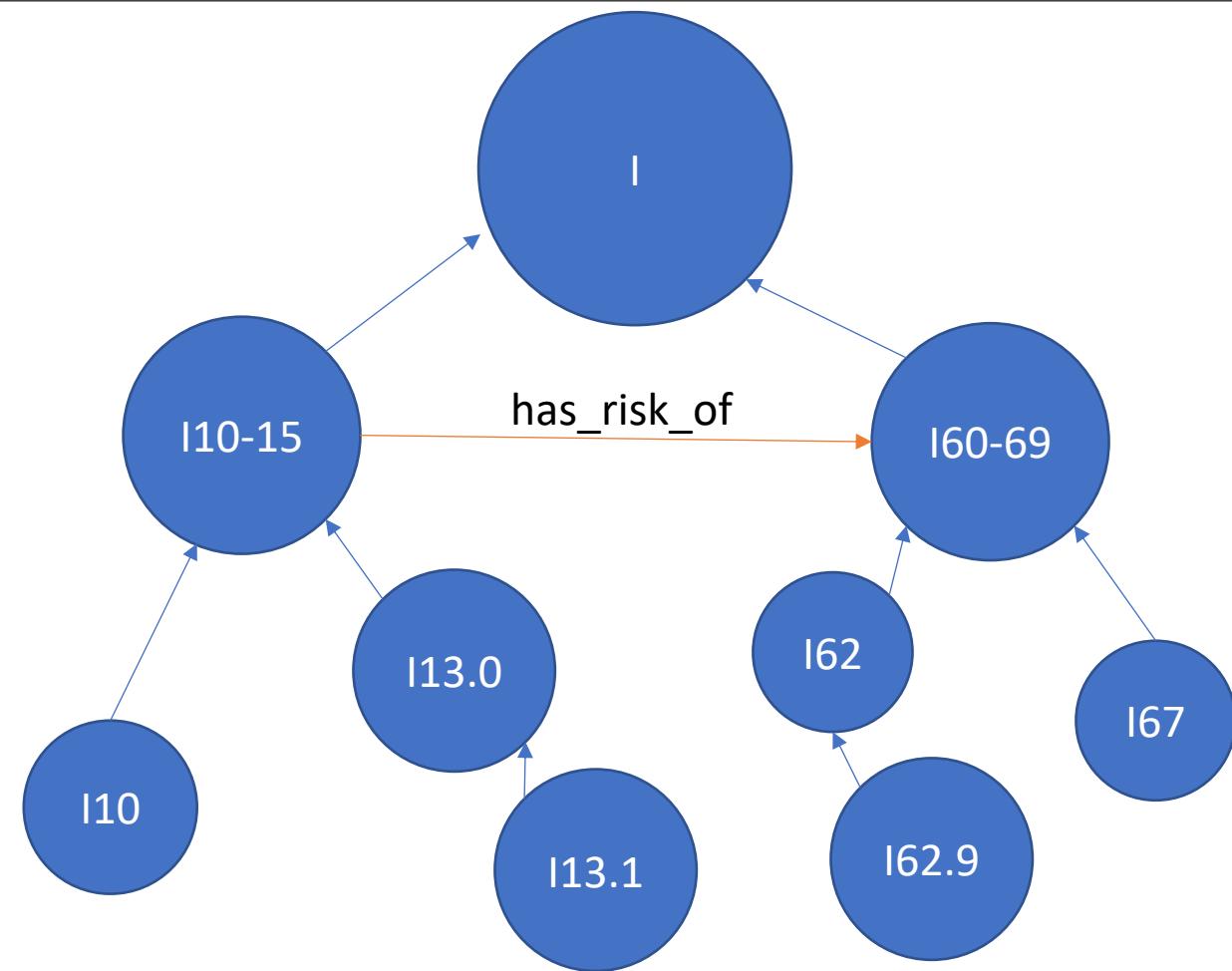
- Complex, relational data models
- Multiple dimensions and links, but little structure to automatically infer relationships and trajectory



Building Relationships for Health Data

Does patient 1 have a risk of stroke?

Patient ID	Diagnosis 1	Diagnosis 2
1	I13.1	
2	I62.9	I10
3	I10	I67

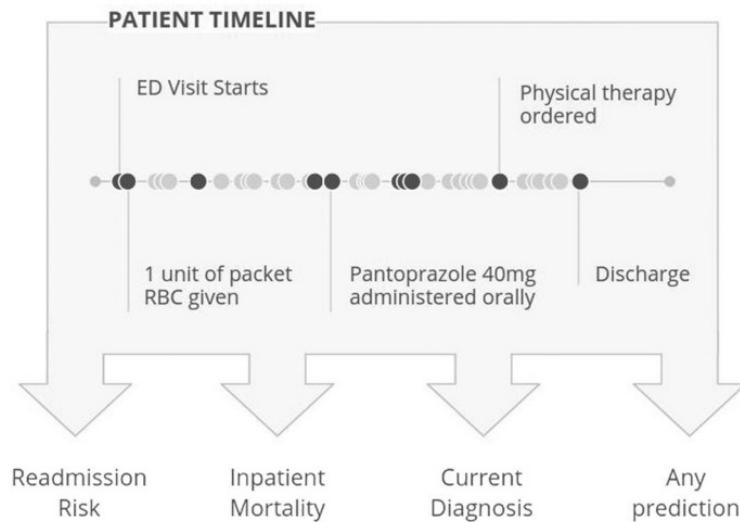


Advanced Models in Healthcare

Scalable and accurate deep learning with electronic health records

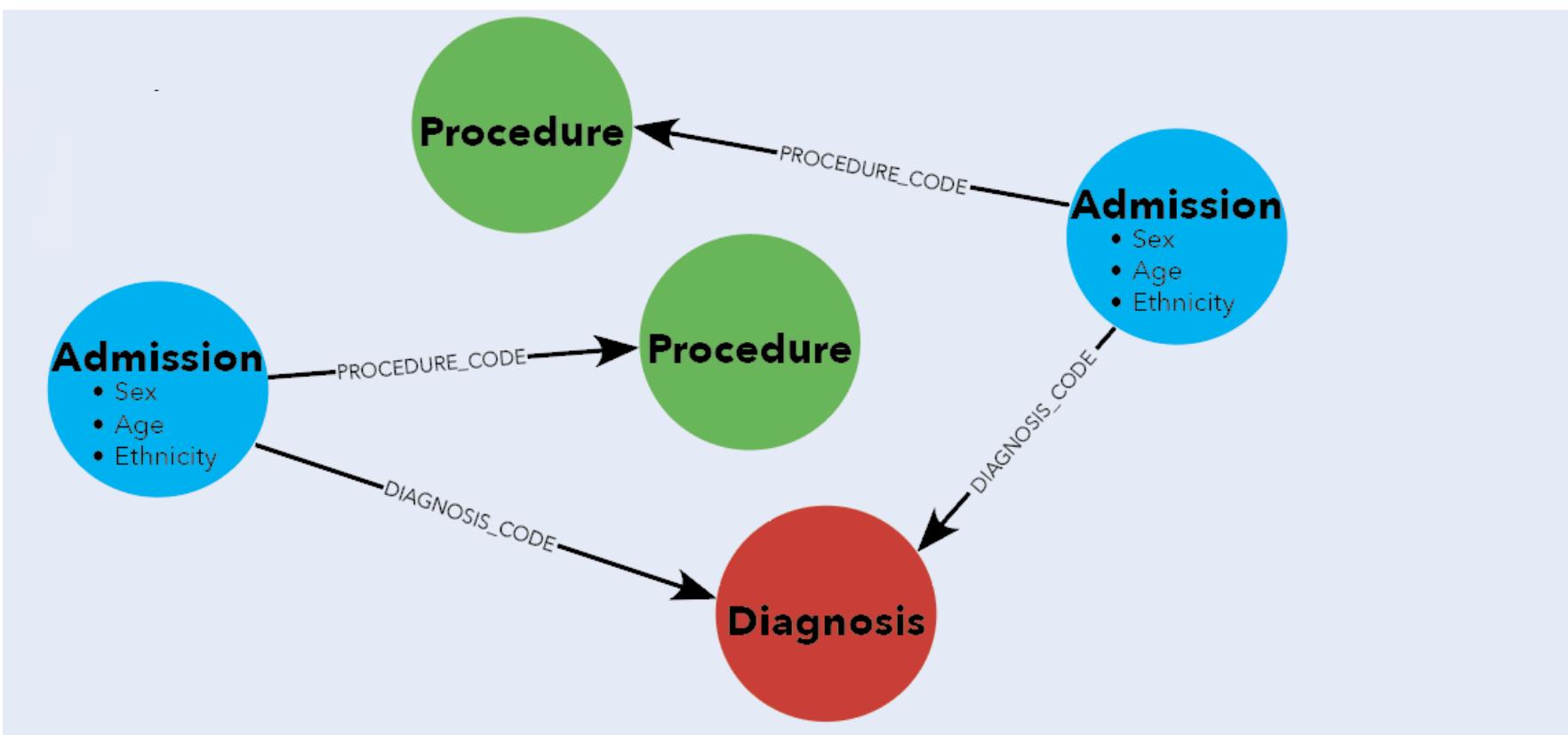
Alvin Rajkomar ^{1,2}, Eyal Oren¹, Kai Chen¹, Andrew M. Dai¹, Nissan Hajaj¹, Michaela Hardt¹, Peter J. Liu¹, Xiaobing Liu¹, Jake Marcus¹, Mimi Sun¹, Patrik Sundberg¹, Hector Yee¹, Kun Zhang¹, Yi Zhang¹, Gerardo Flores¹, Gavin E. Duggan¹, Jamie Irvine¹, Quoc Le¹, Kurt Litsch¹, Alexander Mossin¹, Justin Tansuwan¹, De Wang¹, James Wexler¹, Jimbo Wilson¹, Dana Ludwig², Samuel L. Volchenboum³, Katherine Chou¹, Michael Pearson¹, Srinivasan Madabushi¹, Nigam H. Shah⁴, Atul J. Butte², Michael D. Howell¹, Claire Cui¹, Greg S. Corrado¹ and Jeffrey Dean¹

<https://www.nature.com/articles/s41746-018-0029-1.pdf>



- Patient data from two hospitals in FHIR format, with each resource tokenized
- Each token is embedded, then concatenated for a single embedding per patient
- Trained three time-aware neural networks on four supervised tasks at three temporal points each

Easier with Graphs?



MIMIC in a Graph Model

 Database

 Credentialed Access

MIMIC-IV

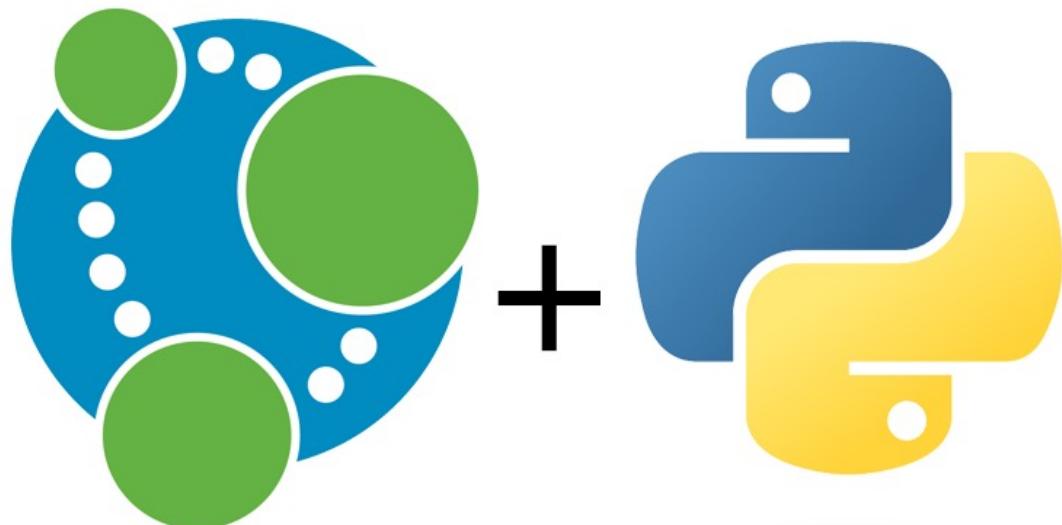
Alistair Johnson  , Lucas Bulgarelli  , Tom Pollard  , Steven Horng  , Leo Anthony Celi  , Roger Mark 

Published: June 12, 2022. Version: 2.0

<https://physionet.org/content/mimiciv/2.0/>

- Semi-public data set
- Requires online/free privacy training and data use agreement
- Access to broad healthcare data from Beth Israel hospital in Boston, MA

MIMIC in a Graph Model



- Many graph database technologies, languages, and libraries to support graph modeling and AI applications
- For online code, using Neo4J and Python with data from MIMIC 3 (ICU data set) to predict length of stay



Modeling and Loading Data

```
from neomodel import StructuredNode, StringProperty, ArrayProperty  
from neomodel import RelationshipTo, RelationshipFrom
```

You, 1 second ago | 1 author (You)

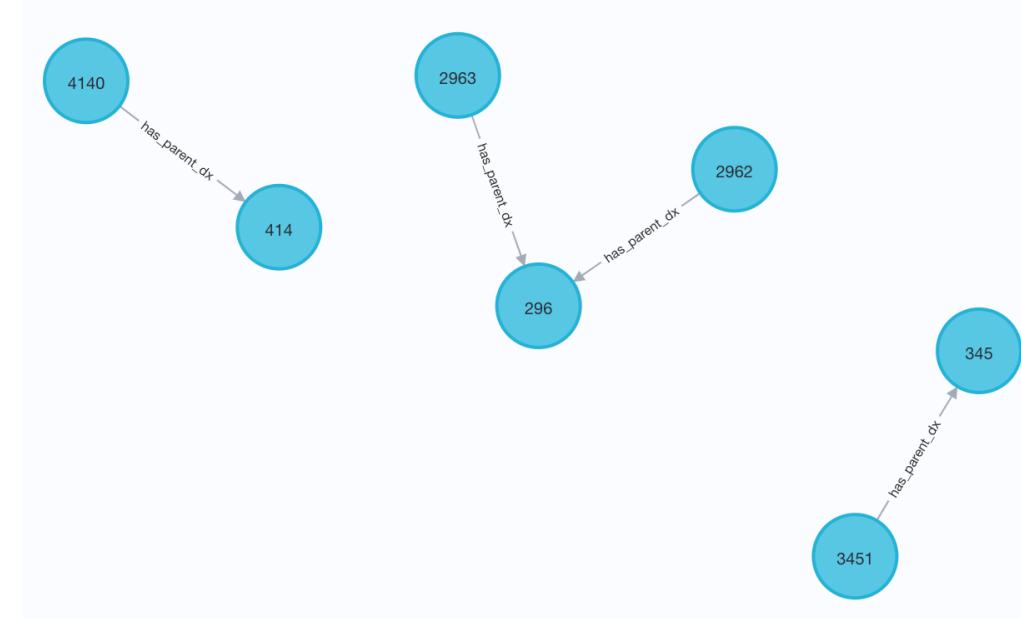
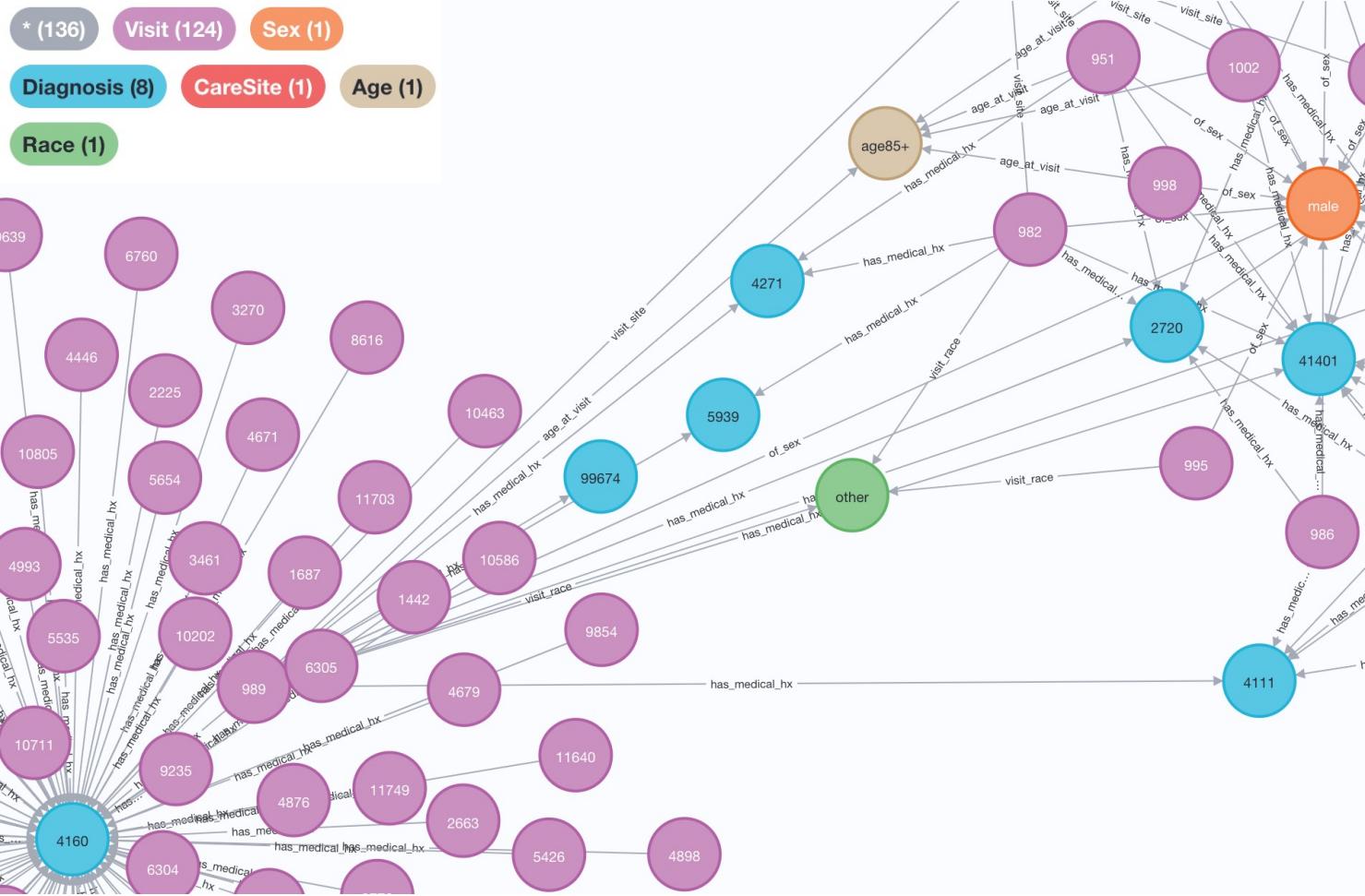
```
class Visit(StructuredNode):  
    visit_id = StringProperty(unique_index=True)  
    embedding = ArrayProperty()  
  
    sex = RelationshipTo("Sex", "of_sex")  
    care_site = RelationshipTo("CareSite", "visit_site")  
    race = RelationshipTo("Race", "visit_race")  
    age = RelationshipTo("Age", "age_at_visit")  
  
    dx = RelationshipTo("Diagnosis", "has_medical_hx")
```

You, 5 days ago | 1 author (You)

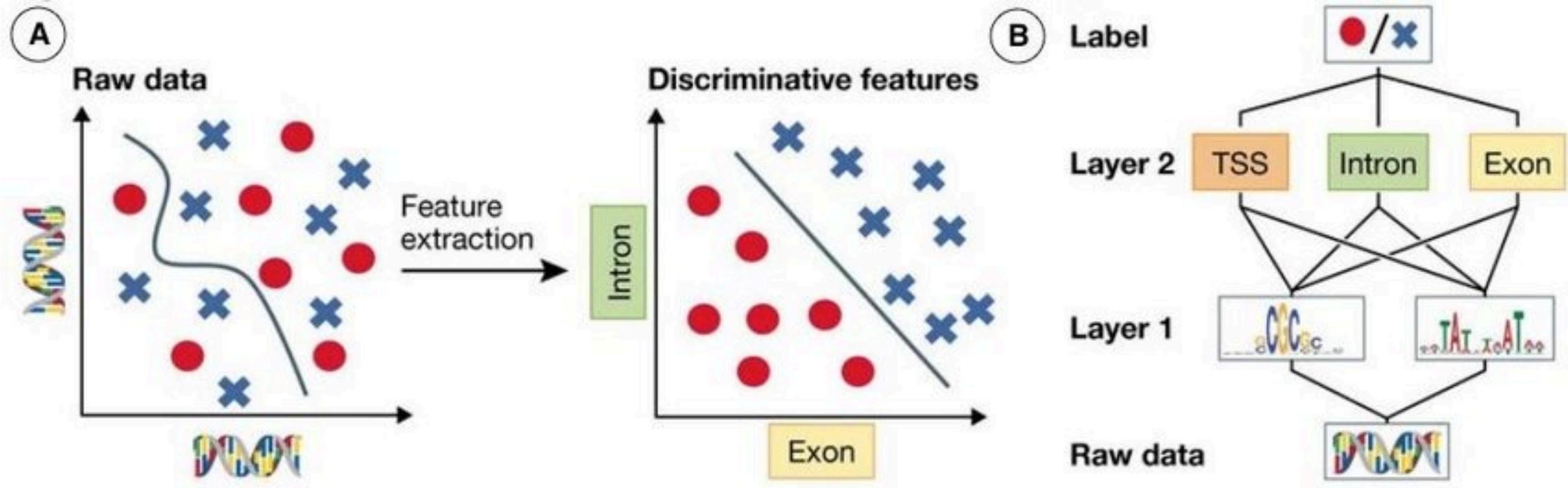
```
class Diagnosis(StructuredNode):  
    icd = StringProperty(unique_index=True)  
    embedding = ArrayProperty()  
  
    child_dx = RelationshipFrom("Diagnosis", "has_parent_dx")  
    parent_dx = RelationshipTo("Diagnosis", "has_parent_dx")  
    visits = RelationshipFrom("Visit", "has_medical_hx")
```

- neomodel: Equivalent of relational database ORM – write model in code, map to graph

Modeling and Loading Data

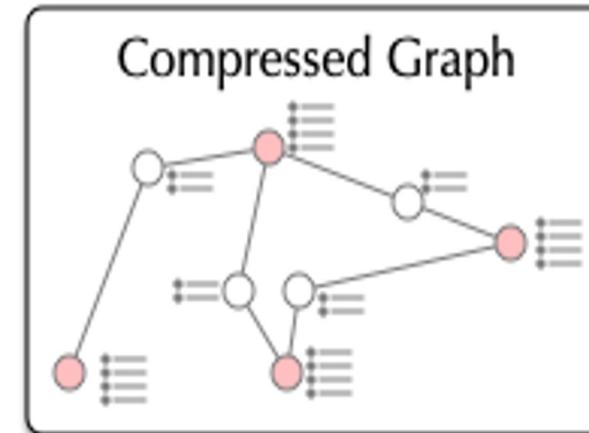
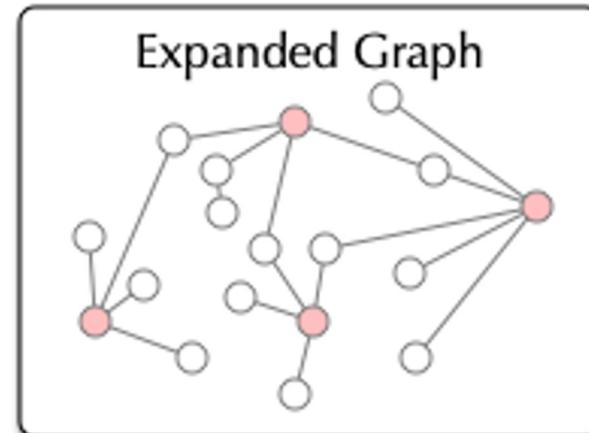


Efficiently Learning a Graph via Representation Learning



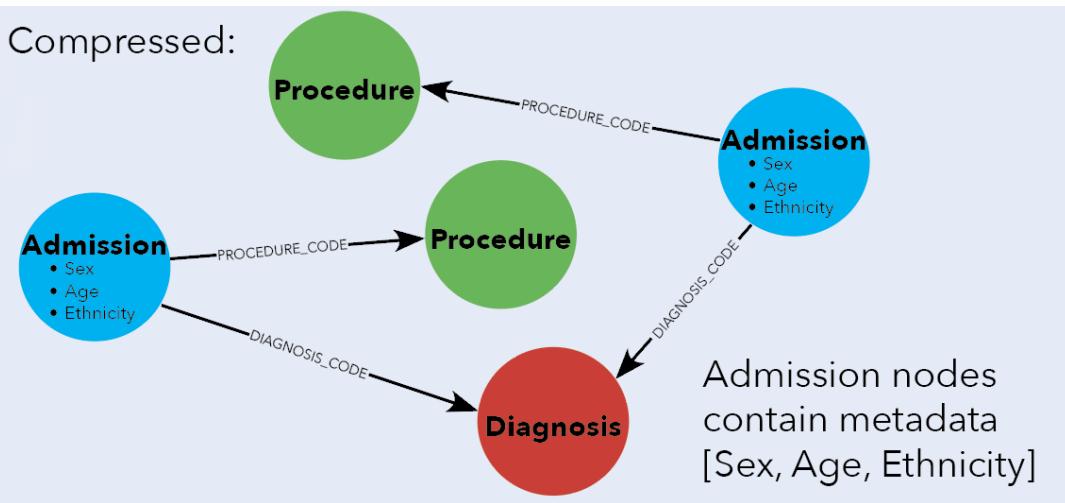
Question: Does our Graph Model Matter?

- One goal is to reduce the need for manual feature engineering
- But really, the goal is to shift some domain expertise effort from the data scientist to the data architect
- With representation learning and graph embeddings – choices in creating our data model might impact downstream predictive performance

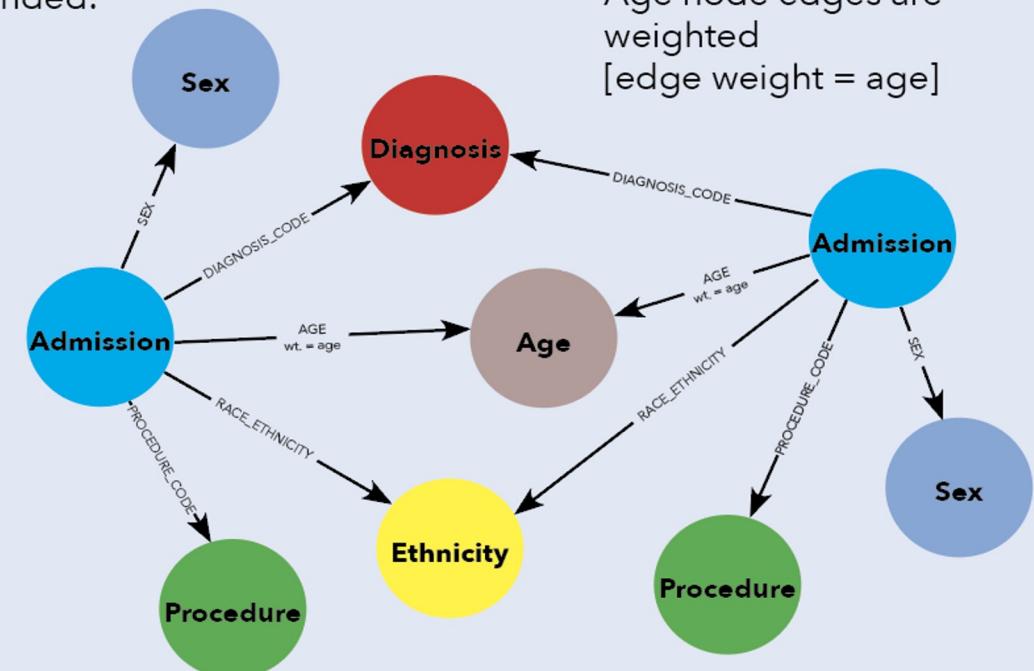


Question: Does our Graph Model Matter?

Compressed:



Expanded:



Question: Does our Graph Model Matter?

- Inclusion Criteria:
 - Age on admission (90+ binned to 90)
 - Ethnicity/Race (top three (African American, Hispanic, White, binned Other))
 - Sex (Male/Female)
 - Diagnosis codes (top 100, summarized to the first 3 digits)
 - Procedure codes (top 40, summarized to the first 2 digits)
- Exclusion Criteria:
 - Age <18

Question: Does our Graph Model Matter?

- Embeddings
 - Two embedding models implemented in Neo4J:
 - Node2Vec
 - GraphSAGE
 - Seven embedding sizes:
 - [20, 50, 100, 150, 200, 250, 300]
 - Two graph directions:
 - Directed, Undirected
- Predictive Model (80/20 train/test split)
 - Two predictive model architectures:
 - Random Forest
 - Logistic Regression
 - Outcome:
 - Length of Stay (<6 days/>6 days)

Using Neo4J GraphDataScience

```
print("Creating projected graph")
G, _ = gds.graph.project(
    'mimic',
    ['Visit', 'Sex', 'Race', 'Diagnosis', 'CareSite', 'Age'],
    ['age_at_visit', 'has_medical_hx', 'has_parent_dx', 'of_sex', 'visit_race', 'visit_site'],
    nodeProperties=["degree"]
)

print("Training GraphSAGE")
model, _ = gds.beta.graphSage.train(
    G,
    modelName = "mimicModel",
    learningRate = lr,      You, 3 days ago • refactor ...
    epochs = 100,
    featureProperties = ["degree"]
)

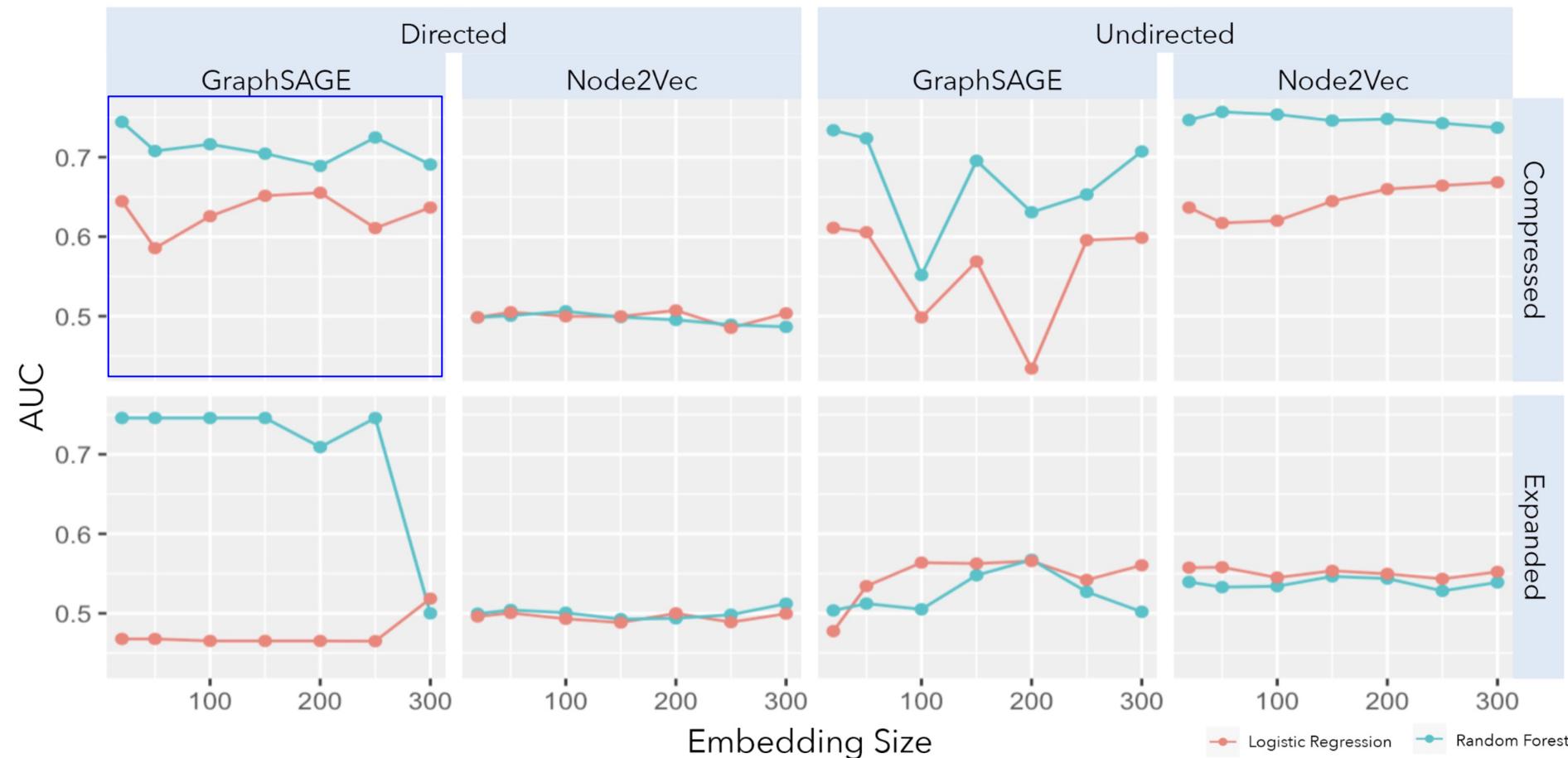
print(model.metrics())

gds.beta.graphSage.write(
    G,
    writeProperty='embedding',
    modelName='mimicModel'
)
```

MIMIC LOS Population Summary

	LOS <6 days	LOS 6+ days	Overall ^{n (%)}
n	17,959	25,999	43,958
Sex M	10212 (56.9)	14639 (56.3)	24851 (56.5)
Age			
18-39	2538 (14.1)	2163 (8.3)	4701 (10.7)
40-51	2842 (15.8)	3470 (13.3)	6312 (14.4)
52-65	4960 (27.6)	7192 (27.7)	12152 (27.6)
66-76	3522 (19.6)	6355 (24.4)	9877 (22.5)
77-84	2404 (13.4)	4391 (16.9)	6795 (15.5)
85+	1693 (9.4)	2428 (9.3)	4121 (9.4)
Race & Ethnicity			
African American	1756 (9.8)	2284 (8.8)	4040 (9.2)
Hispanic	724 (4.0)	868 (3.3)	1592 (3.6)
Other	2885 (16.1)	4188 (16.1)	7073 (16.1)
White	12594 (70.1)	18659 (71.8)	31253 (71.1)

Graph Model and Prediction Accuracy



Graph Model Summary

- GraphSAGE > Node2Vec
 - Overall trend on either graph type or direction
 - Some approaches performed poorly, e.g., Node2Vec on directed graphs (possibly because it hinders the random walk)
- Compressed > Expanded
 - Due to the age node being a central hub that every admission connects to?
- Embedding size - minimal to no impact
- Random Forest vs Logistic Regression
 - RF better: 48%
 - LR better: 12.5%
 - Equivalent: 39.5%

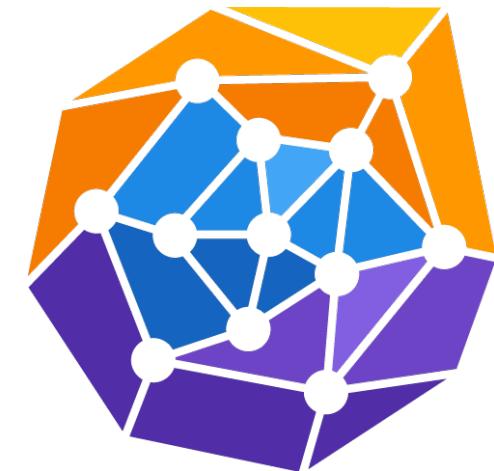
Next Steps

- Convert age nodes to categorical buckets in expanded model (instead of one node with edge weights)
- Use knowledge graphs
 - Adding connections between all ICD code options (Procedure, Diagnosis)
 - from UMLS or OMOP concept hierarchies
 - ICD codes have notion of ordered diagnosis -> current graph with unconnected ICD codes cannot tell connection
 - Primary diagnosis not currently separated out or weighted from secondary diagnoses

Communicating Graph Results

- Graph modeling can present additional challenges in communicating results compared to tabular data given the complexity of models and additional layers to the ML lifecycle
- Important to:
 - Document data sources
 - Describe population data, outcome frequency, and both data modeling and predictive modeling choices
 - Tooling exists to support logging of these data

Open-Source Tools for Graph AI



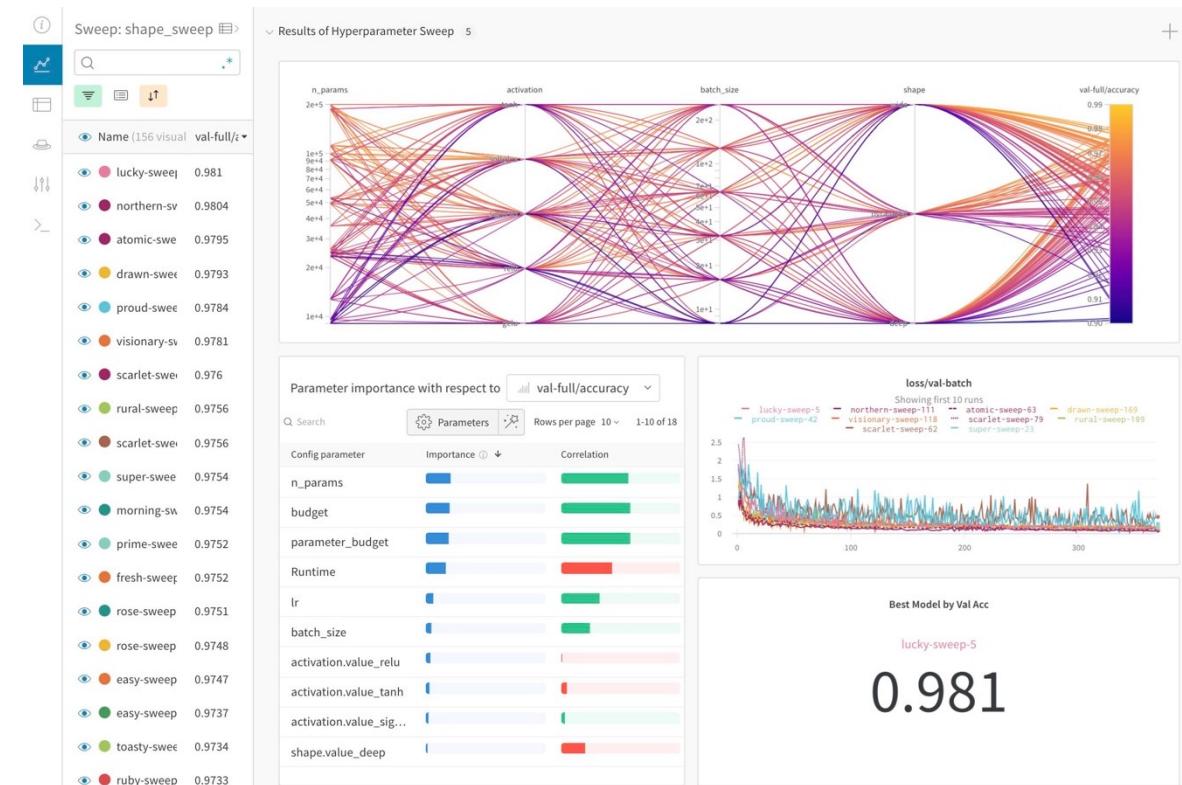
- PyTorch Geometric
 - Library built upon PyTorch to easily write and train Graph Neural Networks (GNNs) for a wide range of applications related to structured data
 - <https://pytorch-geometric.readthedocs.io/en/latest/>
- GraphGym
 - Platform for designing and evaluating GNNs
 - <https://arxiv.org/pdf/2011.08843.pdf>

<https://pytorch-geometric.readthedocs.io/en/latest/notes/graphgym.html>

Open-Source and Commercial MLOps



<https://pytorch-geometric.readthedocs.io/en/latest/notes/graphgym.html>



Conclusions

- Graph networks and graph AI is a rapidly growing field across academia and industry
- Graph models are complex and require new toolsets to efficiently train, evaluate, and use AI models
- Many industry use cases for graph networks, even within specialties in healthcare
- Important to tune not just hyperparameters, but also the data model when using graphs – these choices can impact downstream model performance
- Complexity of these models requires good documentation of data, parameter selection, and performance – visualizations help and tools exist to support this

Acknowledgements

- Sarah Dudgeon – PhD Student
- Katrin Hansel – Postdoctoral Associate
- Fred Warner – Associate Research Scientist
- Patrick Young – Associate Research Scientist
- Andreas Coppi – Associate Research Scientist
- Sameer Pandya – Data Scientist

Healthcare Predictive Modeling with Graph Networks

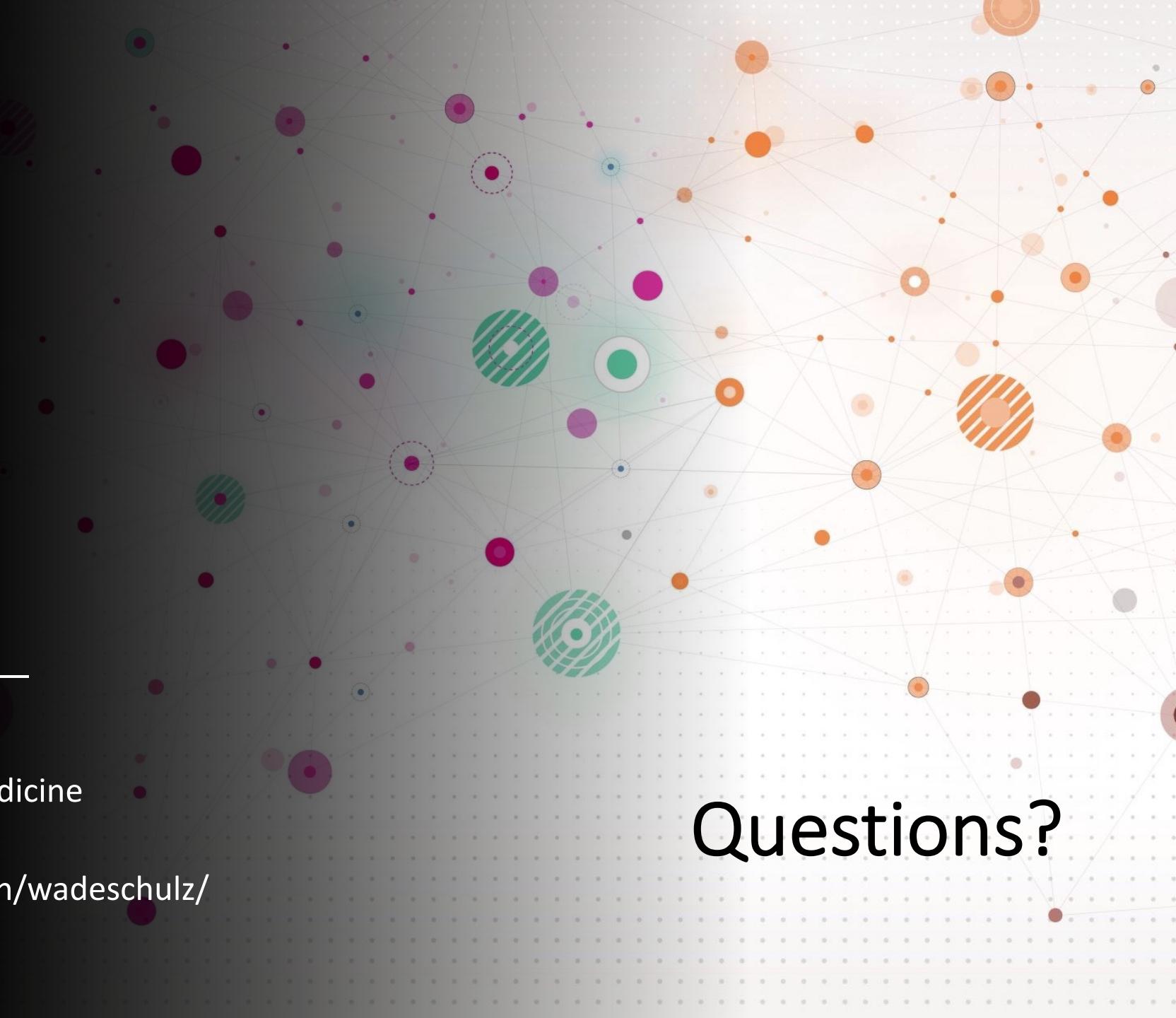
Wade Schulz, MD, PhD

Assistant Professor, Yale School of Medicine

Founder, Refactor Health

LinkedIn: <https://www.linkedin.com/in/wadeschulz/>

Twitter: @wade_schulz

A complex network graph serves as the background for the slide. It consists of numerous small, semi-transparent colored dots (purple, blue, green, orange) connected by thin gray lines, representing nodes and edges in a graph. In the center-right area, there are several larger, more prominent nodes. One node is teal with a diagonal striped pattern. Another is light green with concentric circles. A third is orange with horizontal stripes. These central nodes are interconnected with thicker gray lines, suggesting they are key hubs or entities in the network.

Questions?